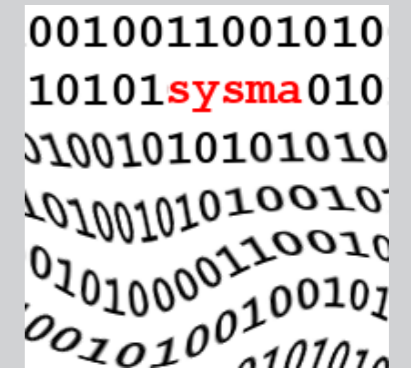


FROM VERIFICATION TO SYNTHESIS OF COMMUNICATION CENTERED SYSTEMS USING BEHAVIORAL TYPES

Hugo Torres Vieira
SYSMA - IMT LUCCA



Software bugs call for verification...

9/9

0800 Antan started
 1000 stopped - antan ✓
 1300 (033) MP-MC 1.582147000 9.037 846 895 console
 (033) PRO 2 2.130476445 4.615925059(-2)
 console 2.130476445
 Relays 6-2 in 033 failed special speed test
 in relay 11.00 test.
 (Relays changed)
 1100 Started Cosine Taps (Sine check)
 1525 Started Multi Adder Test.
 1545 Relay #70 Panel F (moth) in relay.
 First actual case of bug being found.
 Antan started.
 1700 closed down.



Is Knight's \$440 million glitch the costliest computer bug ever?

By Brian Patrick Eha @CNNTech August 9, 2012: 10:22 AM ET

Recommend 154



Facebook Fixes Bug That Exposed Private Chats

By Jeff Bertolucci, PCWorld

May 5, 2010 12:10 PM



Another day, another Facebook security snafu. The popular social network has patched a major security bug that allowed users to snoop on their friends' private chats, and view their pending friend requests.

The exploit caused Facebook to temporarily disable chat, which was back online as of 11 a.m. (US Pacific) on Wednesday.

Users could inadvertently activate the security breach, which was first reported by [TechCrunch Europe](#), via their privacy settings. Facebook has since patched the bug, and sent *PC World* this statement explaining the mishap:

"For a limited period of time, a bug permitted some users' chat messages and pending friend requests to be made visible to their friends by manipulating the 'preview my profile' feature of Facebook privacy settings. When we received reports of the problem, our engineers promptly diagnosed it and temporarily disabled the chat function. We also pushed out a fix to take care of the visible friend requests which is now complete."

Facebook did not report the number of users who were affected by the bug.

The chat exploit is the latest in a long string of Facebook security controversies. Two months ago, the site was victimized by [five different exploits](#), including four [hoax applications](#) and a variant of Koobface virus.

U.S. Senator Charles Schumer (D-N.Y.) last week urged the Federal Trade Commission to create [privacy guidelines](#) for



B MULTIBAND

Reboot and Select proper Boot device or Insert Boot Media in selected Boot device and press a key.

A problem has been detected and windows to your computer.
 KERNEL_STACK_IMAGE_ERROR
 If this is the first time you've seen this message, restart your computer. If this screen appears again, follow these steps:
 Check to make sure any new hardware or software is properly installed. If you need to use Safe Mode to remove or reinstall hardware, press F8 to select the Advanced Boot Options menu.
 Technical information:
 *** STOP: 0x00000077 (0xC0000056, 0xC0000000) ***
 Beginning dump of physical memory

LH	1182	Zurich	06:38
LH	1200	Basle	06:40
LH	124	Stuttgart	06:41
LH	170	Berlin-Tegel	06:41
LH	072	Dusseldorf	06:51
LH	002	Hamburg	07:00
LH	094	Munich	07:00
LH	986	Amsterdam	07:00
LH	1026	Paris-Ch.De Gaulle	07:00
LH	1232	Vienna	07:00
LH	800	Stockholm	07:00
LH	1346	Warsaw	07:00
LH	812	Gothenburg	07:10
LH	1124	Barcelona	07:10
LH	1094	Toulouse	07:20

Single Room No. of rooms: 1

Name: Last name:

<http://www.palacehotel.co.rs>

Molimo Vas da popunite sledeća polja:

- Upišite Vaše ime
- Pogrešno ste uneli telefon
- Upišite Vašu email adresu

OK

Send request

... **but catching bugs is not enough!**

“Don’t try to **frighten** people anymore with the terrible things that are going to happen because it is very ineffective”

[Sir Tony Hoare @ MilnerSymposium2012]

(he said he tried for 20 years without success)



- How to disseminate our theories into engineering practice:

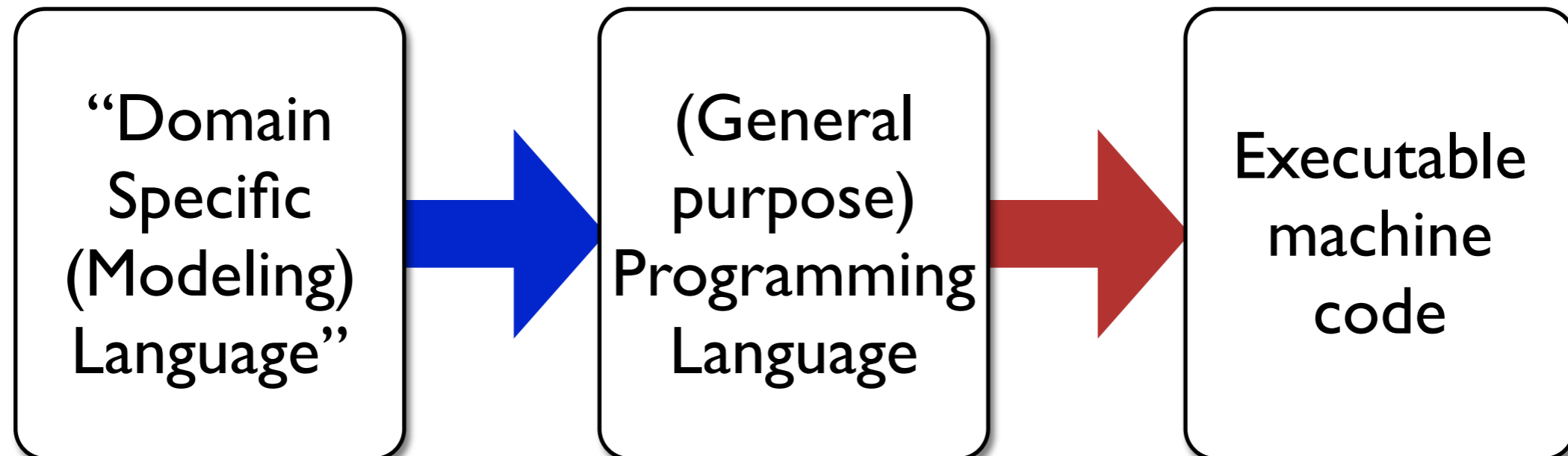
Develop (e.g., program analysis) tools based on our ideas that allow to make more **effective** and **reliable** software in a **cheaper** way

Provide ways to **save money, speed up development, and reduce delivery time** is how our methods can be conveyed to practice

Compilers, PLs, DSLs and beyond

(GP)PLs \Rightarrow Assembly/Machine code (compilers):

- Ambiguity in (natural language) specs
- Lots of recent work on “correctness”

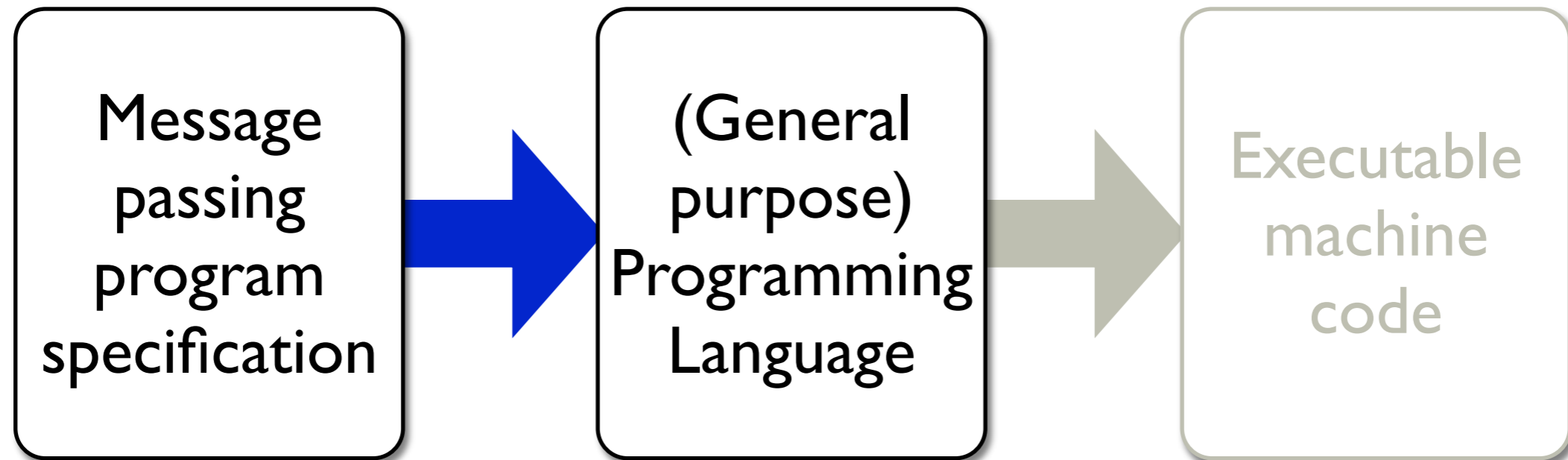


DS(M)Ls \rightarrow (GP)PLs

Although DSLs have been widely used to speed up development time there is not so much supporting (theoretical) work

Communication centered systems

Lots of (theoretical) work
on models and specifications



DS(M)Ls -> (GP)PLs

Although DSLs have been widely used to speed up development time there is not so much supporting (theoretical) work

Derivation, verification and beyond

$$\frac{\begin{array}{c} \vdots \\ \hline p_1 \end{array} \quad \dots \quad \begin{array}{c} \vdots \\ \hline p_k \end{array}}{\Gamma \vdash P}$$

Relation between types (specifications) and programs

Derivation, verification and beyond

$$\frac{\begin{array}{c} \vdots \\ \hline \rho_1 \end{array} \quad \dots \quad \begin{array}{c} \vdots \\ \hline \rho_k \end{array}}{\Gamma \vdash P} \quad \text{Relation between} \\
 \text{types (specifications)} \\
 \text{and programs}$$

`bool type_check(type_spec, program, ...)`
 $\Gamma \vdash P$

`type_spec type_inference(program, ...)`
 $P \Rightarrow \Gamma$ (such that $\Gamma \vdash P$)

Derivation, verification and beyond

$$\frac{\begin{array}{c} \vdots \\ \hline \rho_1 \end{array} \quad \dots \quad \begin{array}{c} \vdots \\ \hline \rho_k \end{array}}{\Gamma \vdash P} \quad \text{Relation between} \\
 \text{types (specifications)} \\
 \text{and programs}$$

`bool type_check(type_spec, program, ...)`
 $\Gamma \vdash P$

`type_spec type_inference(program, ...)`
 $P \Rightarrow \Gamma$ (such that $\Gamma \vdash P$)

`program synthesis(type_spec, ...)`
 $\Gamma \Rightarrow P$ (such that $\Gamma \vdash P$ - Any P ?)

Progressing Sessions/Conversations

DLY TGC 07

P PLACES 13

Binary

VV COORDINATION 13

BCD et al. CONCUR 08

Multiparty

CV ESOP 09

CDPY COORDINATION 13

PVV COORDINATION 14

P, Q	$::=$	$\mathbf{0}$	(Inaction)
		$P \mid Q$	(Parallel)
		$(vx)P$	(Restriction)
		$x!y.P$	(Output)
		$x?y.P$	(Input)
		$\star x?y.P$	(Replicated Input)

p	$::=$	$!$	(Output)
		$?$	(Input)
		τ	(Synchronization)
L	$::=$	end	(No interaction)
		$e \ p \ T. \ L$	(Session)
T	$::=$	L	(Linear)
		$e \ p \ T$	(Shared)

- Processes are characterized by their
 - **communication capabilities on each channel**
 - $\Gamma = channel_1 : T_1, \dots, channel_k : T_k$
 - **(overall) communication ordering**
 - $< =$ strict partial order of communication events

$$\Gamma; < \vdash P$$

- the ordering describes the “after than” dependencies among communication events

Theorem (Preservation)

If $\Gamma; < \vdash P$ and $P \rightarrow Q$ then $\Gamma; < \rightarrow \Gamma'; <'$ and $\Gamma'; <' \vdash Q$.

Theorem (Liveness)

Let $\Gamma_1, x: B_1; <_1 \vdash P_1$ and *matched*($\Gamma_1, x: B_1$). If e in *events*(B_1) then there is $P_1 \rightarrow^n P_2$ and $(\Gamma_1, x: B_1); <_1 \rightarrow^n (\Gamma_2, x: B_2); <_2$ for some $n > 0$, such that $\Gamma_2, x: B_2; <_2 \vdash P_2$ and e is not in *events*(B_2).

B_1, B_2 are linear types

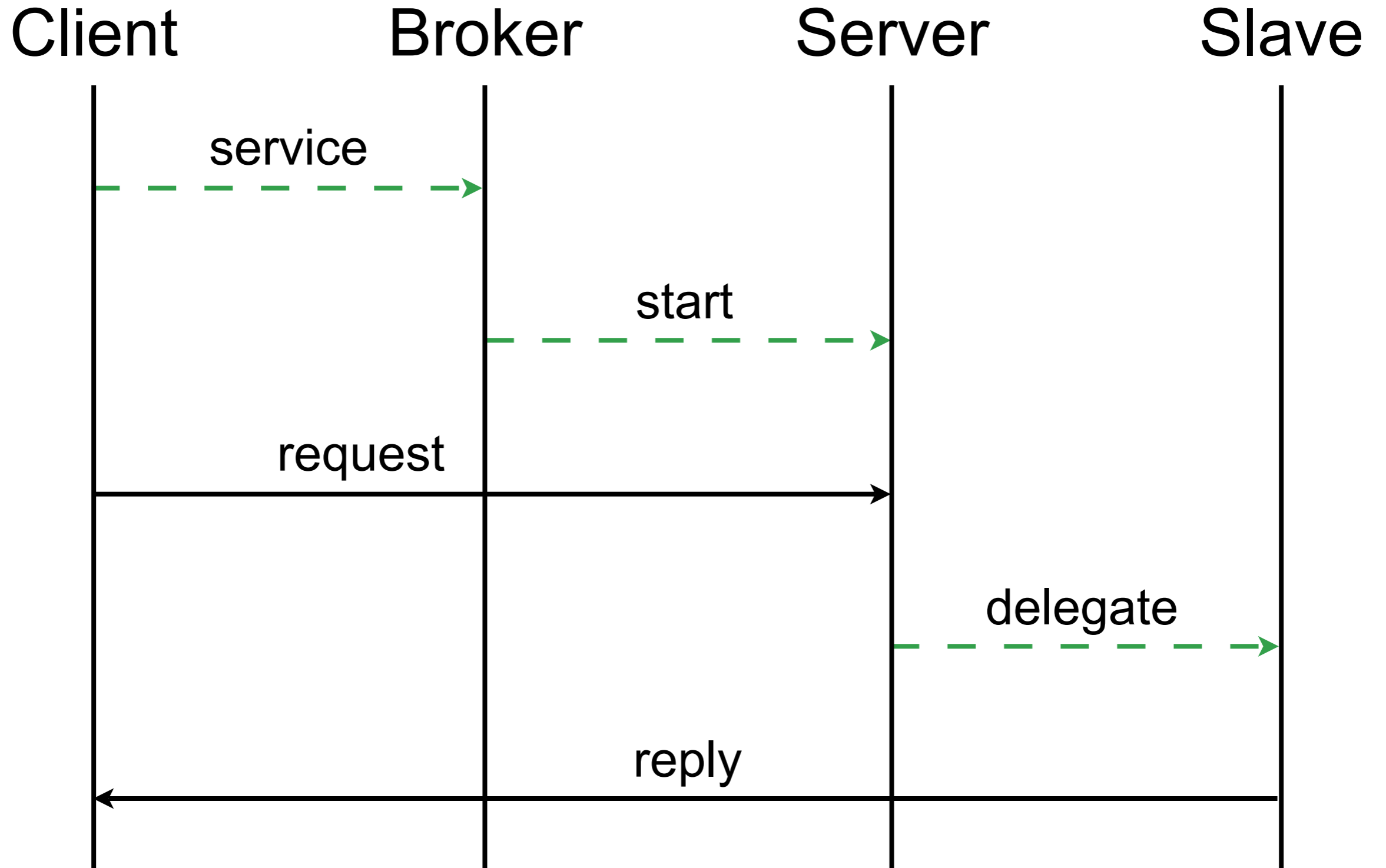
A typing Γ is *matched* if (roughly) it only specifies τ types.

events(B) is the set of events mentioned in type B

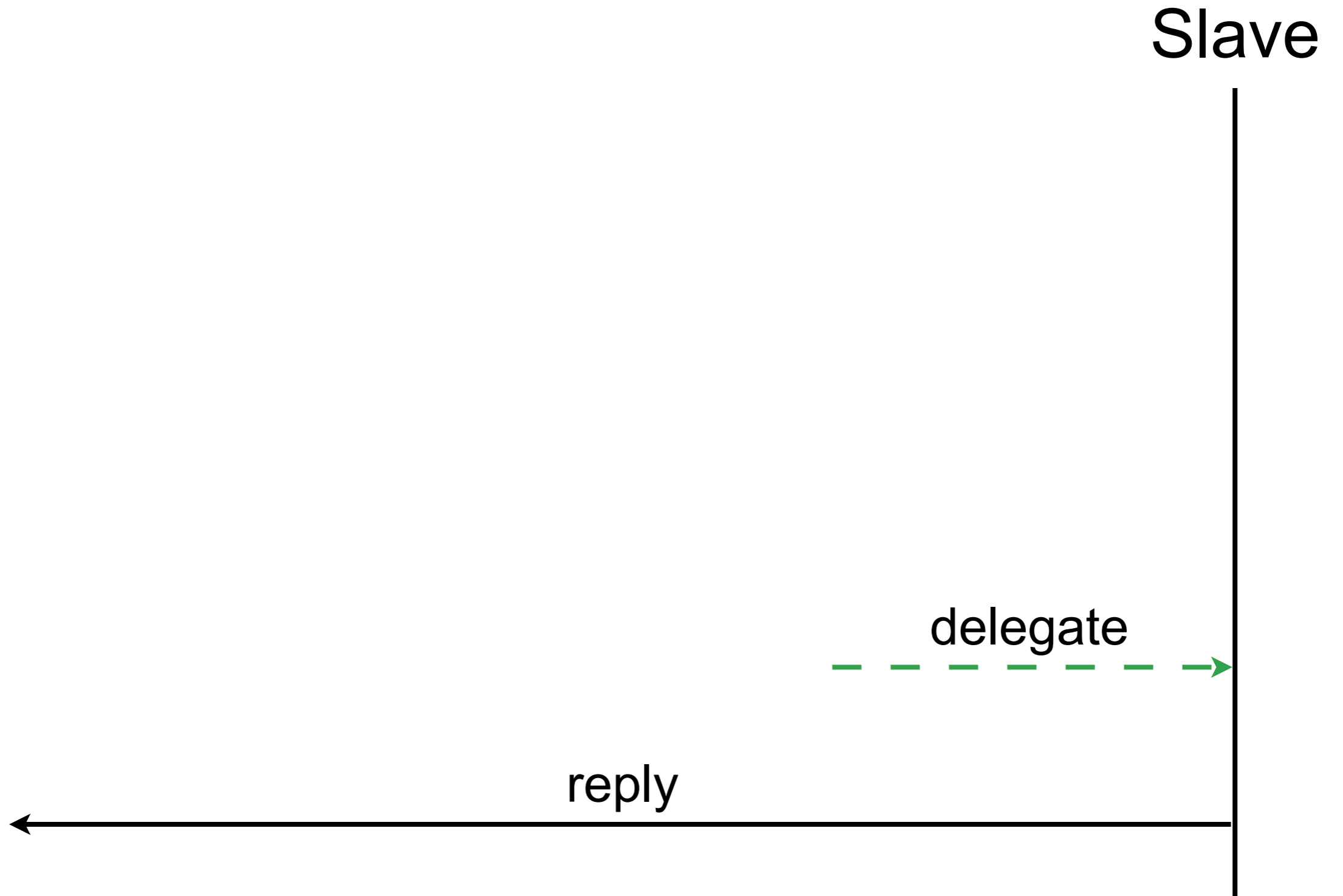
Synthesis algorithm - Principles

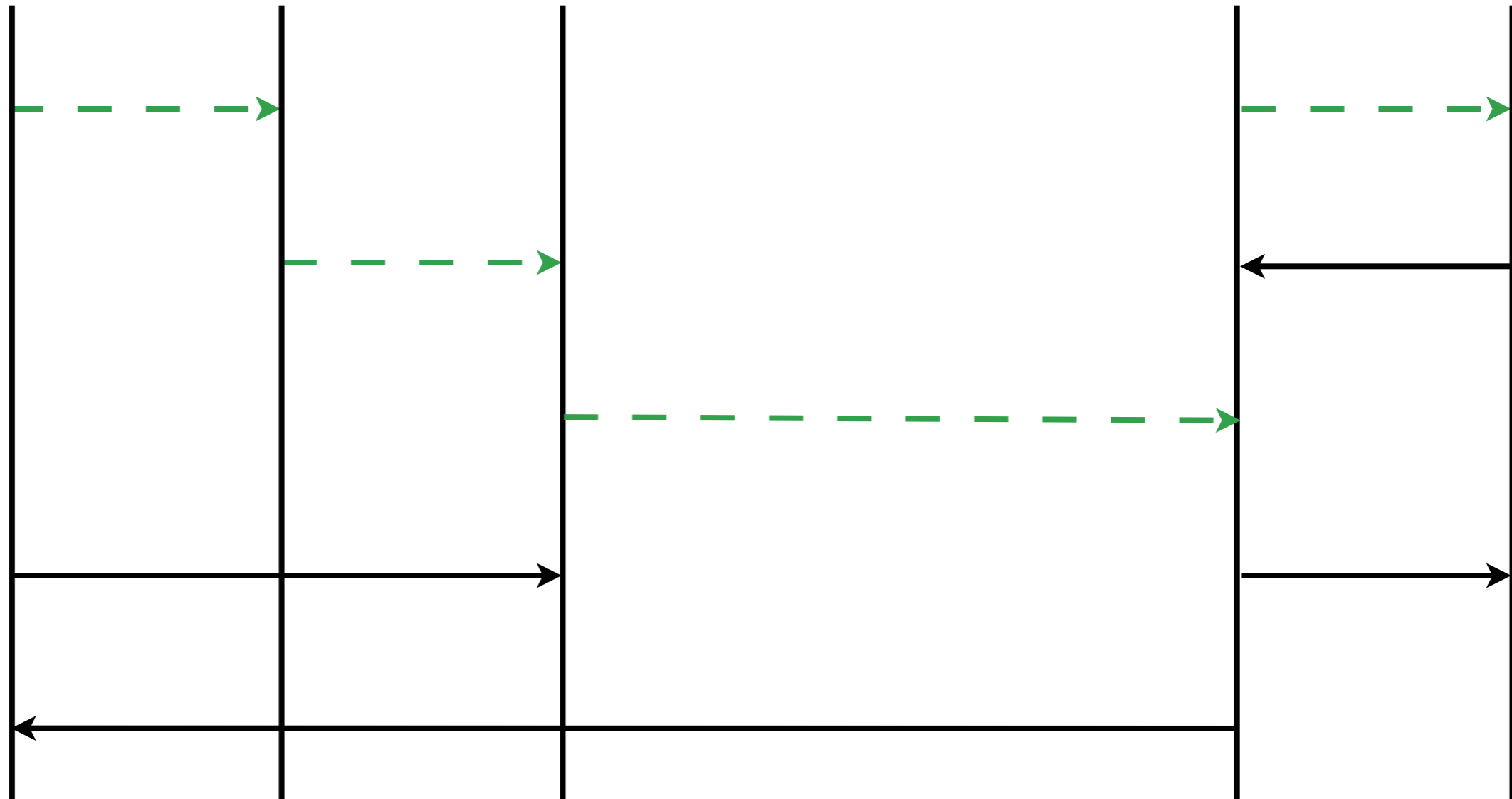
- **Separate synchronization events in the two counterparts**
 - keep relation acyclic considering the unification of the two events but order the individual “polarized” events
- **Explore usages and ordering in a combined way**
 - Synthesize parallel composition whenever order has more than one minimal element
 - If order has one minimal element synthesize the corresponding action prefix
 - continuation is synthesized using events of greater order
 - for outputs if the carried type is “present” in the environment use the respective name, otherwise generate new name (restriction)

Message sequence chart - Global view



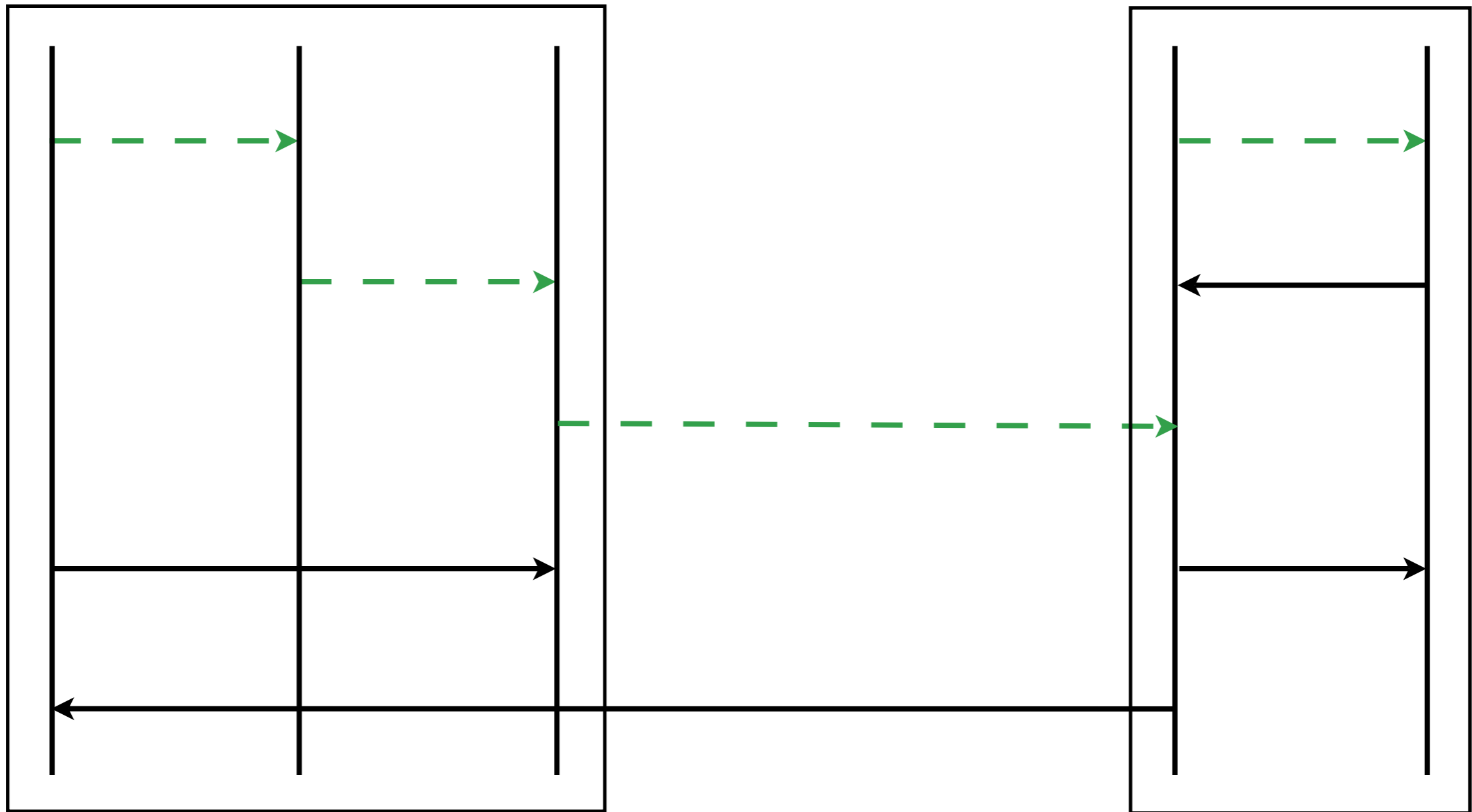
Message sequence chart - local view





S1

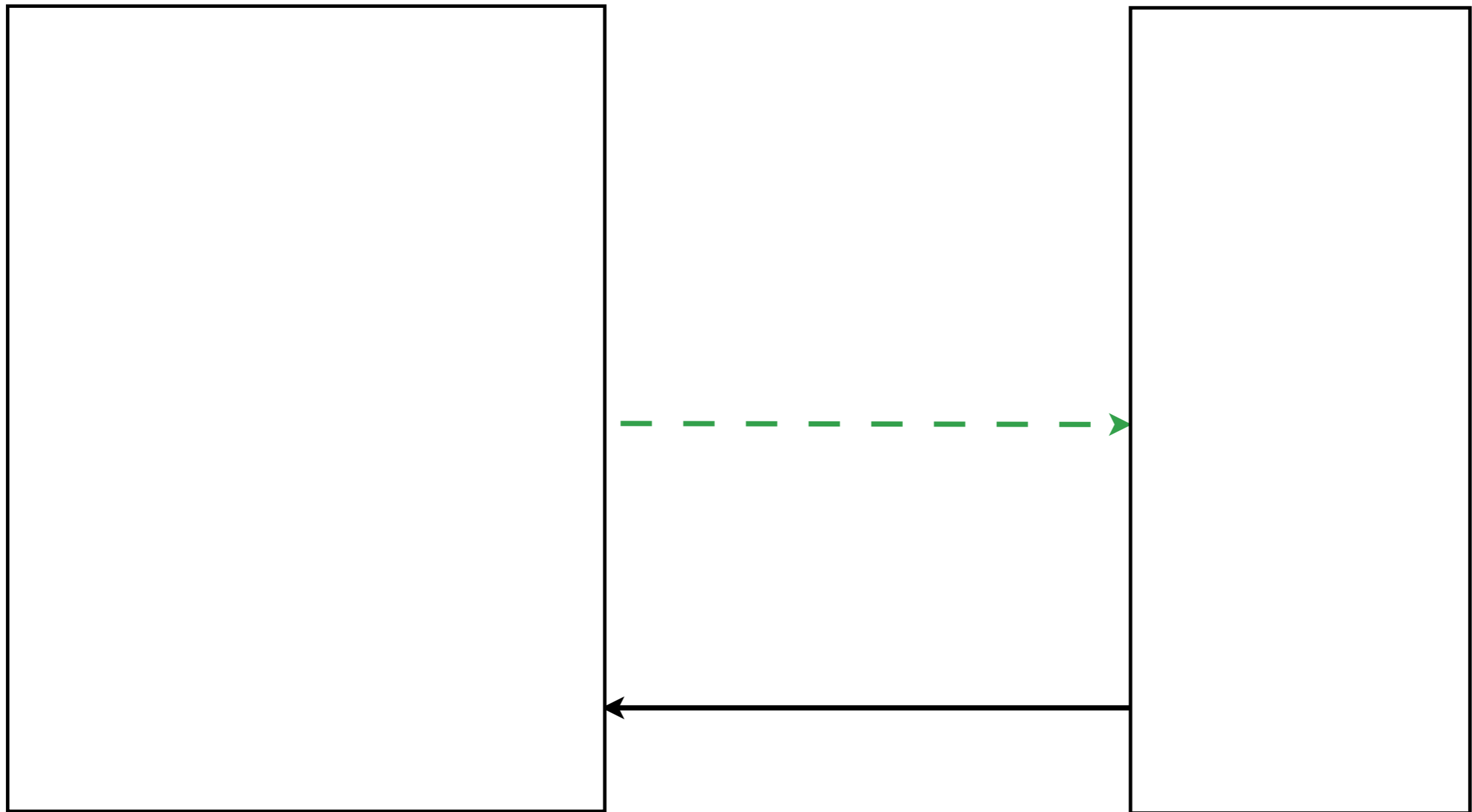
S2



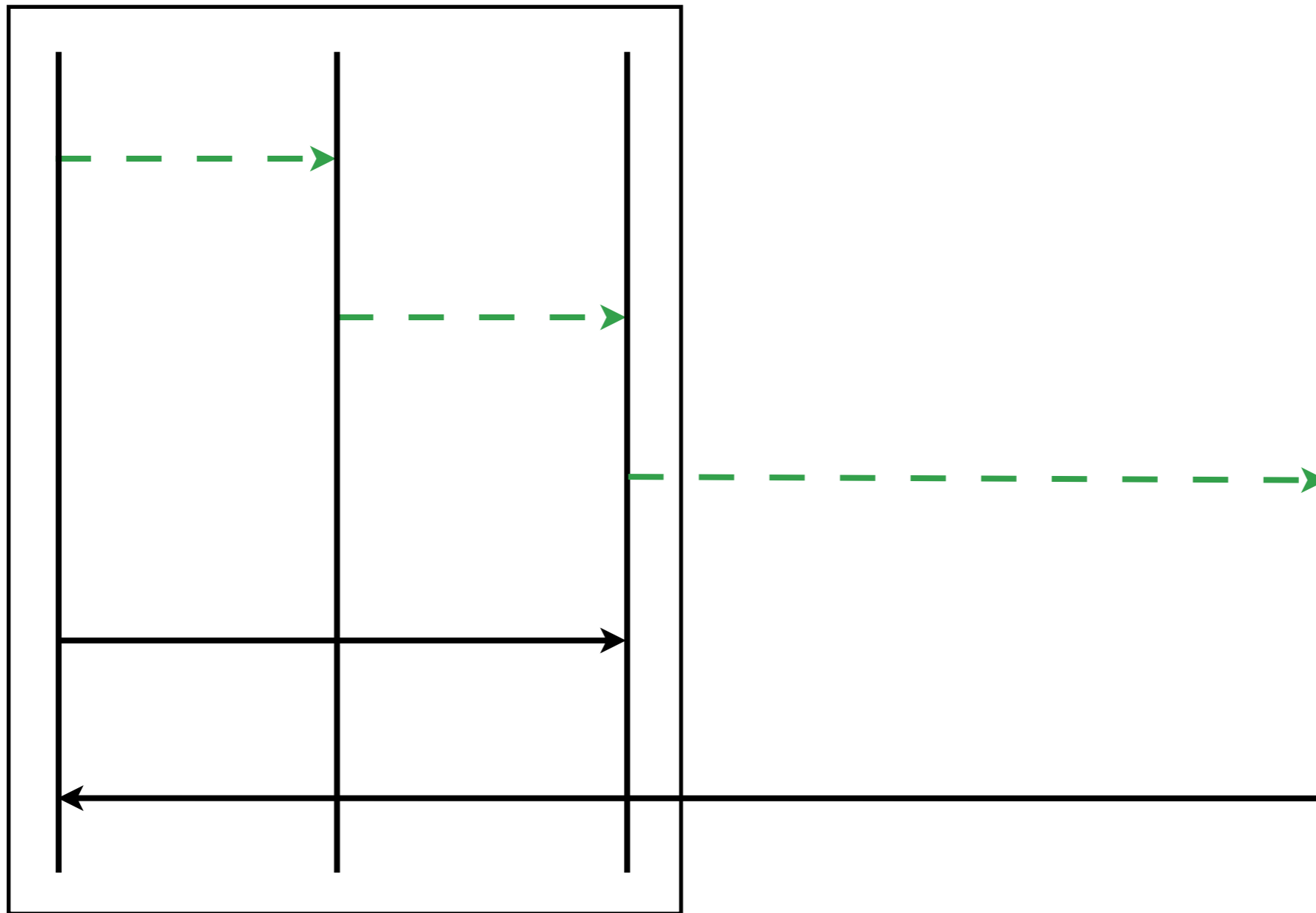
Session Modules

S1

S2



SI



Advantages of type driven synthesis

- **Type safety and other properties ensured by construction**
 - Preventively saving a lot on debugging time
- **Compositionality**
 - Adding features with compatible specs means new code will be compatible
 - Saving a lot on maintenance effort
- **Validation of generated code wrt specification**
 - For instance liveness provides a tight correspondence between system and spec
 - Behavioral equivalences between specification and system
- **Implementation**
 - Bridge the gap between software engineers and formal specification tools